Reinforcement Learning-Driven Locomotion Policy Optimization for R-Hex Robots with Low-Cost Sensor Solutions

Alejandro Jose Agapito
Faculty of Mechanical Engineering
National Engineering University
Lima, Peru
alejandro.agapito.q@uni.pe

Juan Salvador Palacios
Faculty of Mechanical Engineering
National Engineering University
Lima, Peru
juan.palacios.b@uni.pe

Miguel Angel Chumacero
Faculty of Mechanical Engineering
National Engineering University)
Lima, Peru
miguel.chumacero.b@uni.pe

Fabrizio Salomon Castellares
Faculty of Mechanical Engineering
National Engineering University
Lima, Peru
fabrizio.castellares.h@uni.pe

Julio Cesar Casquero
Faculty of Mechanical Engineering
National Engineering University
Lima, Peru
jcasquero@uni.edu.pe

***Abstract*— This paper introduces a novel approach to optimize locomotion control for R-Hex robots using reinforcement learning (RL) with low-cost sensor configurations. The approach aims to address the needs of smallholder farmers in Peru by providing a resilient, affordable robotic solution adaptable to challenging terrains, such as steep slopes and rocky soils. The prototype will be evaluated through rigorous testing in simulated and real agricultural environments, with a focus on terrain adaptability, control precision, and cost-efficiency.**

Keywords*— *RHex Robot, AIOT, reinforcement learning, policy training, cloud.

I. Introduction

Agricultural areas in Peru face numerous topographical challenges—including steep slopes, sandy and rocky soils, and uneven terrain—that hinder the use of conventional agricultural robotics. These geographic limitations significantly restrict the navigation, stability, and operational capabilities of traditional robotic systems, posing barriers to tasks such as soil preparation, planting, and monitoring. Furthermore, limited financial resources among Peruvian smallholder farmers, with an annual average income of approximately USD 6,500, exacerbate these challenges by impeding the adoption of advanced agricultural technologies [1].

This paper proposes an innovative approach employing reinforcement learning (RL) algorithms tailored to the needs of agricultural robots operating in complex terrains. By utilizing RL, we aim to streamline robotic motion control systems, potentially reducing dependency on costly encoders and sensors that are commonly required for navigation and stabilization in rough environments [2]. We focus on the R-Hex robot platform, selected for its minimal engine requirements and off-road capability, as a testbed for examining the viability of RL-controlled robotics in rural agriculture.

Our proposed approach seeks to create a cost-effective, resilient robotic solution accessible to smallholder farmers, enhancing affordability and functionality. RL algorithms, which adapt and improve through repeated interactions, offer an adaptive control solution that aligns well with the unpredictable terrains found in Peruvian agriculture, potentially reducing equipment costs while enhancing robot resilience [3].

To validate this approach, we will develop a prototype of the R-Hex robot equipped with RL-based controls. The prototype will undergo rigorous testing in both simulated and real-world agricultural environments, with performance metrics including terrain adaptability, control precision, and cost-efficiency. This iterative testing will facilitate refinements to the RL algorithms, ensuring that the final design meets the unique demands of Peru's diverse agricultural landscapes.

II. R-Hex Mechanical and Sensor System of Use

A. Motor and Sensor Selection

To reduce implementation costs, inexpensive and commercially available electronic components were selected. Actuators were selected based on their capabilities to meet the needs of operation in harsh agricultural environments, eliminating advanced components such as encoders, and replacing them with more affordable and efficient sensors.

Due to the absence of encoders in the motors, position and stability monitoring of the robot was implemented using proprioceptive sensors. Therefore, the MPU6050 gyroscope and accelerometer were integrated together with an FC-51 infrared sensor, forming a cost-effective monitoring solution that meets the requirements for stability and navigation on uneven terrain.

B. Mechanical Structure and Design Optimization

The hexapod design of the R-Hex robot, inspired by insect locomotion, presents significant challenges in terms of stability and control. To manage these challenges, SolidWorks was used to create a detailed 3D model that includes every component and joint, allowing for a thorough evaluation of its structure and functionality.

The main structure was optimized to provide the necessary rigidity and stability in difficult terrain. The robot's legs were designed using a geometry based on the Lamé ellipsoid, ensuring a balanced distribution of loads during movement. This configuration allows for precise and agile locomotion on uneven terrain, as shown in Fig. 1.

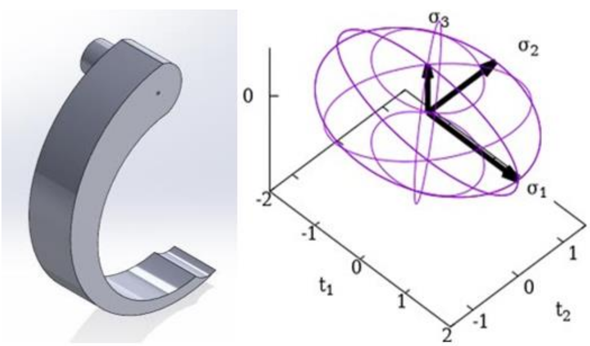


Fig. 1. Design of the R-Hex leg with the Lamé ellipsoid, optimizing load distribution.

The complete assembly of the robot is shown in Fig. 2, where the integrated arrangement of the components in a compact and functional structure can be seen, thus ensuring that the robot meets the requirements of durability and efficiency in agricultural applications.

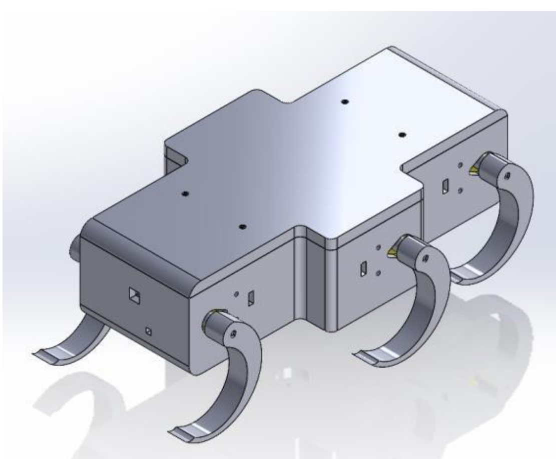


Fig. 2. Complete assembly of the R-Hex, showing the optimized structure for stability and maneuverability in difficult terrains.

C. Schematic and Circuit Designal Structure and Design Optimization

The design of the R-Hex electronics was developed in EasyEDA Pro, optimizing the connections of all electrical components. In the PCB design process, power and signal transmission requirements were studied, assigning track widths that optimize functionality and minimize interference, as shown in the schematic diagram of the system (Fig. 3).

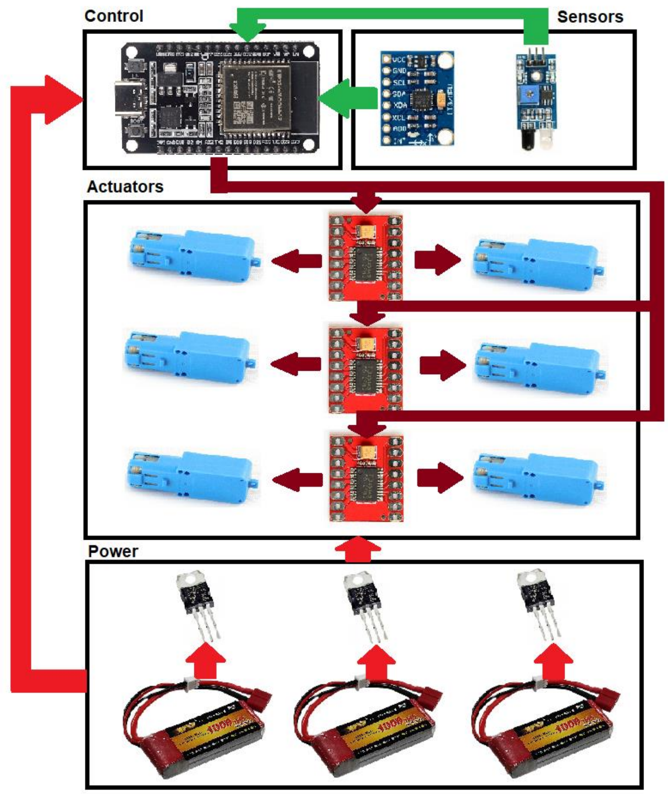


Fig. 3. Robot pictorial diagram.

The EasyEDA Pro program was used to develop the schematic diagram, which made it possible to organize the respective connections of all electrical components of the R-Hex.

D. PCB Design for Compact Integration

The PCB was configured with an optimized size to integrate into the central body of the robot, minimizing the space used without compromising power and data transmission capability. The routing was designed to support both data transmission and power transmission, achieving a compact and efficient design (Fig. 4).

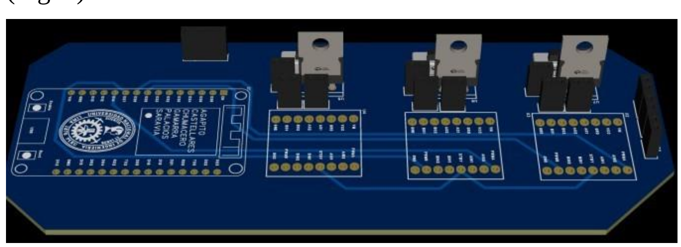


Fig. 4. 3D PCB design for the R-Hex in EasyEDA Pro.

III. TECHNIQUE IN ROBOT KINEMATICS

A. R-Hex Robot Leg Trajectory Planning

The walking trajectory of the RHex robot is based on a locomotion pattern that encompasses several modes, including Partial Wave, Alternating Tripod, Full Wave, Single and Tetrapod. This locomotion strategy provides remarkable stability during movement.

In Alternating Tripod Mode, the left front and rear legs move simultaneously along with the middle right leg. Subsequently, the other three legs move. This gait pattern, common in many hexapods (subphylum of arthropods), allows for greater stability by keeping the center of gravity within the triangle formed by the three legs in contact with the ground. Also, this gait mode prevents slippage, since, while three legs are in motion, the other three are elevated.

This approach not only minimizes the risk of tipping, but also ensures continuous and safe movement of the RHex robot on uneven terrain. In order to improve overall stability and reduce the impact of leg collisions with the ground, specific path planning methods have been implemented.

During linear displacement, the robot legs follow a precise sequence, which is illustrated in Figure 5. This figure shows the position of the body and legs along the motion sequence, thus ensuring a smooth and controlled gait.

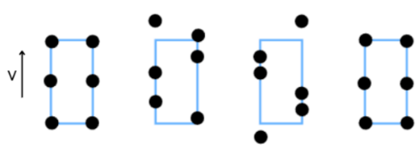


Fig. 5. Position of the foot and body when walking in static gait.

B. Robot State Analysis

To optimize its performance, the RHex robot operates in different states: "Standing", "Forward" and "Backward". Each state presents specific challenges, and reinforcement learning algorithms are employed to adapt to the environments. Key aspects of each state are detailed below, along with settings that allow the robot to maintain stability and reduce power consumption. Figure 6 shows the running state and static state positions.

- Standing State : In the "Standing" state, the RHex robot remains static and in equilibrium, without any movement. This state is crucial for defining a stable posture from which the robot can start or stop any movement, especially in complex terrain. With reinforcement learning, the robot adjusts the angle and position of each leg in response to perturbations, thus achieving a stable posture with minimal energy and stress on the joints.

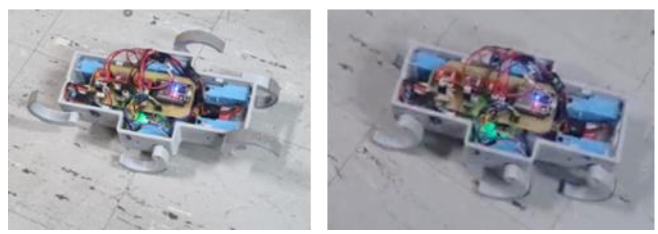
  This state also allows the robot to conserve energy and minimize mechanical wear while evaluating the support surface. Thanks to learning algorithms, the robot learns posture patterns to adapt to different surfaces, reinforcing optimal positions to maintain balanced support on inclined or uneven terrain.

- Forward State: In the "Forward State", stability control is essential to avoid falling or slipping when facing obstacles or slopes. Reinforcement learning allows the robot to adapt to terrain conditions, coordinating the movement of each leg to maximize traction and reduce errors. The reward function focuses on the robot's speed and stable posture during forward movement.

  This model also considers the slope of the terrain, adjusting the length and frequency of each step to maximize travel efficiency, ensuring accurate and continuous progress.

- Backward State: The "Backward State" presents an additional challenge due to limited visibility and the need to fine-tune the direction. In this state, the RHex robot adjusts the force and direction of each leg to move backward in a controlled manner, with the reward function focusing on stability and lateral displacement control.

  This state is useful in escape maneuvers or positional adjustments, where the robot moves backward quickly and safely without compromising its structural integrity or that of the environment. The learning model ensures trajectory correction and reinforces stability in real time.



(a) (b) Fig. 6. States of the RHex robot. (a) Running state of the RHex robot. (b) Static state of the RHex robot.

IV. AI SYSTEM DESING

Our system is designed to enable efficient reinforcement learning for locomotion on real-world hardware, prioritizing flexibility and robustness in unstructured environments. Unlike many studies that rely on pre-defined motion primitives and external motion capture systems, our approach allows the robot to learn autonomously using only onboard proprioceptive sensors. This choice aligns with our goal to facilitate training in various outdoor environments without requiring specialized instrumentation.

To achieve this, we use a low-level action space that operates directly on the robot's pulse-width modulation (PWM) signals. Additionally, real-time processing and control are achieved through wireless communication between the robot's microcontroller unit (MCU) and a remote processing computer. In the following subsections, we detail the design choices for state and action spaces, as well as our approach to reward engineering, aimed at producing stable and adaptive locomotion behaviors in the wild.

A. Reinforcement Learning Overview

Reinforcement learning (RL) provides a robust framework for training agents to perform tasks by interacting with an environment. Through trial and error, an RL agent learns to optimize its policy based on feedback from rewards and penalties, ultimately striving to maximize cumulative rewards over time. This process is especially suited to dynamic and complex environments, as RL can enable robots to adapt to unstructured terrains without extensive pre-programming. However, traditional RL methods often face challenges in sample efficiency and high data requirements, especially in real-world robotics applications. As advancements such as model-free RL techniques evolve, the barriers related to sample inefficiency in RL applications are being addressed.

B. Model-Free Reinforcement Learning and Sample-Efficient Algorithms

In model-free RL, agents learn policies directly through interaction, without relying on a pre-built model of the environment. This approach can simplify implementation and reduce computational overhead, making it a strong candidate for real-world robotic applications. Sample-efficient algorithms within model-free RL, such as Soft Actor-Critic (SAC) and its variants, have proven particularly useful for robotics, as they allow rapid learning with fewer environmental interactions. Building on this, prior research (e.g., Smith et al.) demonstrated that, with precise system design and algorithm tuning, quadruped robots could achieve stable locomotion on diverse terrains in as little as 20 minutes of direct real-world training [4]. This finding underlines the viability of training robots in natural environments, potentially bypassing the need for extensive simulation phases.

C. Algorithm Selection

In selecting an RL algorithm for this work, we aimed to balance sample efficiency with computational feasibility, ensuring rapid learning and policy convergence directly in real-world environments. For this purpose, Dropout Q-learning (DroQ) was chosen due to its proven effectiveness in achieving high sample efficiency and stability in continuous control tasks, especially in real-world, unstructured settings.

TABLE I. OVERVIEW OF TRAINING HYPERPARAMETERS

Hyperparameter	Value
Discount Factor γ	0.99
Critic Learning Rate	0.001
Actor Learning Rate	0.0003
Learnig Start Steps	1000
Batch Size	256
Update-to-Data Ratio (UTD)	20

D. State and Action Spaces

The design of state and action spaces is crucial for the RL agent's ability to learn effective control policies. In this work, the state space is composed of parameters critical to the robot's stability and orientation, including:

- Root orientation: This includes roll and pitch to monitor the overall stability of the robot.
- Root angular velocities: Capturing the angular velocity around roll, pitch, and yaw axes provides feedback on balance and rotational stability.
- Z-position with ground reference: This serves as a reference for the robot's height relative to the ground, which is essential for traversing uneven terrain.
- Current PWM motor signals: These signals are vital for the real-time monitoring and adjustment of motor power output.

The action space, in contrast, involves the variations in PWM signals, constrained within a range of ±8. These PWM adjustments control the robot's movement direction and speed, with absolute PWM values restricted between 80 and 180 to avoid dead zones and minimize abrupt movements. This constrained action space, normalized between -1 and 1, enhances control precision and safety, particularly when operating on irregular terrain.

E. Reward Function Design

The reward function is an essential component of the RL algorithm, driving the agent's learning by associating actions with positive or negative outcomes. In this system, the reward is formulated as an exponential function based on the error between reference and current states. The general form of the reward function is:

Where a and b are parameters controlling the reward magnitude and sensitivity to error, respectively. The appendix provides the specific values used for these parameters, enabling fine-tuning of the reward's responsiveness to the robot's state accuracy.

F. Hardware Setup

The training was conducted using the SBX framework, a custom reinforcement learning platform based on Stable Baselines3 and implemented in JAX for computational efficiency. This setup ran on an HP Pavilion 15 laptop equipped with an NVIDIA GeForce GTX 1650 GPU, providing the necessary processing power for efficient DroQ updates in real-time field settings.

The training frequency was set at 20 Hz, allowing for synchronous state updates and action predictions with each interaction step. Communication between the laptop and the on-board ESP32 microcontroller (MCU) was maintained via Wi-Fi, with state data transmitted from the MCU and action commands sent back for immediate execution. The ESP32 managed real-time control by adjusting the PWM motor signals based on the RL policy's output, ensuring responsive and smooth operation across various terrain types. This configuration, combining the computational capabilities of the HP Pavilion with the low-latency control of the ESP32, enabled effective training directly in real-world environments, facilitating an adaptable, portable learning system suitable for dynamic field conditions.

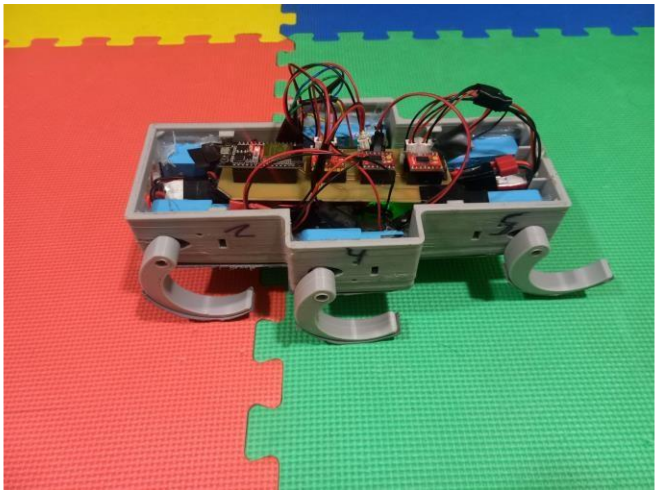


Fig. 7. R-Hex training environment

V. Conclusions

In conclusion, this study demonstrates the feasibility of an R-Hex type robot controlled by reinforcement learning (RL) algorithms to improve locomotion and stability in complex agricultural terrains. The implementation of a low-cost system based on sensors such as the MPU6050 gyroscope and accelerometer and an infrared sensor has enabled efficient monitoring without the need for expensive encoders. This is especially relevant in the context of agriculture in Peru, where the irregular topography and limited resources of small farmers make accessibility to advanced technologies challenging.

The results of locomotion tests in simulated and real terrain suggest that reinforcement learning can effectively adapt to terrain variations, achieving stability and control in the three main operating conditions: standing, forward and reverse. In the forward state, the RL model optimizes traction and smooth motion, while in the standing and backward states, posture and control accuracy are improved, reducing part wear and maximizing the robot's energy efficiency. This adaptability demonstrates the model's potential to operate in real-world conditions and respond to the needs of farmers in difficult terrain.

In addition, the mechanical design of the R-Hex offers reliable and safe mobility. The integrated sensor and motor architecture allows for an effective balance between cost and performance, a significant achievement for future applications in agricultural monitoring and maintenance. With additional optimizations in the hardware design and the RL model, it is possible to achieve a highly efficient autonomous system with extended autonomy.

The results of this research serve as a basis for future research in low-cost agricultural robotics. The development of a working prototype like the AI and IoT-controlled R-Hex opens up new opportunities to implement robots that benefit smallholder farmers in topographically challenged regions. The next phase of work will include more extensive testing on even more varied terrain and the integration of machine vision models to identify pests and assess crop health, moving towards a comprehensive agricultural robotics solution.

VI. Future Work

In future work, we plan to integrate a plant pest recognition model into the robot, thus extending its functions to agricultural monitoring. It is expected that the robot will improve its performance in hostile terrain through training in real agricultural environments, where terrain conditions can be optimally replicated. This in situ training approach will allow adjusting the control and detection algorithms to the specific characteristics of these environments.

It is advisable to optimize the number of sensors on the robot, as more sensors can reduce autonomy due to additional power consumption. Minimizing the use of redundant sensors will help to extend the operation time and reduce the frequency of battery recharging, thus maintaining the functionality of the robot without compromising its efficiency.

For real-time sensing, the YOLO artificial intelligence model, which has demonstrated high accuracy in computer vision applications, will be used. In particular, the YOLOv5 model will be implemented due to its low computational cost and ease of training in PyTorch [5], which will facilitate its integration into the system. In addition, a Kaggle dataset with preprocessed data will be used to ensure robust training of the model, maximizing its ability to identify pests and optimize the robot's response in the field.

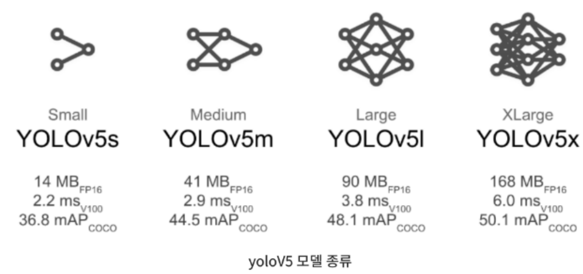


Fig. 8. Versions of YOLOv5

References

[1] N. B. R. Briceño, E. B. Castillo, J. L. M. Quintana, S. M. O. Cruz, and R. S. López, "Deforestation in the Peruvian Amazon: Land cover change indices and land use based on GIS," *Boletín de la Asociación de Geógrafos Españoles*, no. 81, 2019.

[2] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: A survey," in *Proc. IEEE Symp. Series Comput. Intell. (SSCI)*, Dec. 2020, pp. 737-744.

[3] L. Benos, V. Moysiadis, D. Kateris, A. C. Tagarakis, P. Busato, S. Pearson, and D. Bochtis, "Human–robot interaction in agriculture: A systematic review," *Sensors*, vol. 23, no. 15, p. 6776, 2023.

[4] L. Smith, I. Kostrikov, and S. Levine, "A Walk in the Park: Learning to Walk in 20 Minutes With Model-Free Reinforcement Learning," *arXiv preprint arXiv:2208.07860*, 2022. [Online]. Available: https://arxiv.org/pdf/2208.07860

[5] J. Tordesillas Torres, *Design and Simulation of the Locomotion System of a Hexapod Robot for Search and Rescue Tasks*, Bachelor's thesis, Escuela Técnica Superior de Ingenieros Industriales, Universidad Politécnica de Madrid, Madrid, Spain, 2015-2016.

[6] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A Review of Yolo algorithm developments," *Procedia Comput. Sci.*, vol. 199, pp. 1066-1073, 2022.

- **The below form will not be published, but it is necessary to help with the review process.**

Authors' Background

Name	Email	Position (Prof , Assoc. Prof. etc.)	Research Field	Homepage URL
Alejandro José Agapito Quiñones	alejandro.agapito.q@uni.pe	**Student Researcher**	**Mechatronics**	**https://www.linkedin.com/in/alejandro-jose-agapito-qui%C3%B1ones-1424b4238/**
Juan Salvador Palacios Bett	juan.palacios.b@uni.pe	**Student Researcher**	**Mechatronics**	**https://www.linkedin.com/in/juansalvaorpalaciosbett**
Miguel Angel Chumacero Bustamante	miguel.chumacero.b@uni.pe	**Student Researcher**	**Mechatronics**	**https://www.linkedin.com/in/miguel-angel-chumacero-bustamante-b66a01289/**
Fabrizio Salomon Castellares Huaman	fabrizio.castellares.h@uni.pe	**Student Researcher**	**Mechatronics**	**https://www.linkedin.com/in/fabrizio-castellares-huam%C3%A1n-ba24b3238/**
Julio Cesar Casquero	jcasquero@uni.edu.pe	**Mechanical Engineer**	**Mechanics**	**https://www.reasearchgate.net/profile/Julio-Casquero**